Text to Hypertext: Can Clustering Solve the Problem in Digital Libraries?

Robert B. Kellogg

Information Systems
PRC Inc., Reston, VA 22091
kellogg_bob@prc.com

Madhan Subhas

Department of Computer Science
Virginia Tech, Blacksburg, VA 24061
madhan@csgrad.cs.vt.edu

Abstract

Automatic hypertext generation remains an extremely challenging endeavor in the digital library world. In this paper we present a solution for automatically connecting relevant information in dynamic textual digital libraries. This textual information is generally unconnected and often unexplored due to the large flow of information entering from remote and local sources. Often, full-text indexes exist for this information but embedded links to related information are conspicuously absent. Links that do exist are usually generated in an arduous and time-consuming manual process. That is why the ability to automatically generate links has a potentially high payoff.

Our solution for the automatic generation of hypertext links relies on the techniques of document segmentation and document clustering. Hypertext links are automatically generated during the document clustering process using the incremental cover-coefficient-based clustering algorithm. The issues of link completeness and link quality are also addressed in this paper. Link completeness is studied by comparing the cluster-based approach of link generation to the exhaustive link generation approach. Results indicate that links are more complete in the higher similarity range than in the lower similarity range. Initial link quality user studies indicate that the cluster-based hypertext link generation approach is promising. In the future, we plan to conduct further studies on link quality and investigate ways to increase the effectiveness of our approach.

1 Introduction

As the countries of the world compete in an ever expanding global market, information will become the single most important resource for economic growth, national security, and education. Much of this valuable information will be available in the digital libraries of the world due to the continual technological advances in the computer industry and the explosive growth of the Internet. Recent studies on Internet growth have revealed that the World-Wide Web has tripled in size over the seven month period from December 1994 to June 1995 [Bou95]. As this ocean of data continues to expand, and as advances in electronic communication technologies make hypermedia accessibility a reality, new techniques will be needed to seek out and maintain pointers to relevant information. Without the ability to automatically connect relevant information, the strategic use of these libraries will not reach full potential.

The advantages of hypertext as an information retrieval tool are well known. Adding a linking layer on top of existing indexes can improve a user's chances of finding the "right" information. Combining information retrieval techniques with hypertext empowers a user to issue directed content searches or simply browse collections while looking for relevant information. For example, documents returned from a query may contain links to other documents of interest that were missed by the original query.

Automatically generating links between related text is an extremely challenging endeavor. In fact, little progress has been made since Vannevar Bush introduced hypertext to the world. Many projects have attempted to automatically generate links using varying approaches [All95] [Far89] [Tho91] but have met with limited success. Often their approach suffers from tradeoffs such as creation of a system that maximizes effectiveness while placing little emphasis on efficiency, or creation of a system that only works for static libraries, disregarding dynamic digital libraries. The ability to automatically and dynamically generate links between related documents based on a global view of the collection is our goal.

Another significant objective of this project is integration of automatic hypertext linking with current academic and commercial systems including Virginia Tech's Envision digital library and PRC's *Productivity Edge*^{TM} document management product. Envision allows full-text searching and full-content retrieval [Hea95] on a collection of computer science literature. It features a unique visualization method for displaying the results of a query. In the Envision digital library, the automatic linking capability will assist students and teachers with making the most efficient and effective use of the library. *Productivity Edge* is a flexible data management solution designed to effectively manage business documents and engineering drawings – from creation, through revision, to distribution and storage. In *Productivity Edge*, the linking capability will allow users to quickly and easily locate and access dynamic on-line business information.

In this paper, we will discuss our solution to the automatic hypertext link generation problem. We use the techniques of document clustering and document segmentation to solve this problem. The paper is organized in the follow-

DL'96, Bethesda MD USA
© 1996 ACM 0-89791-830-4/96/03..$3.50

ing manner:

1. A brief overview of the current research in the field of automatic hypertext link generation.
2. Our solution to the link generation problem.
3. A discussion of the experiments to test the validity of our approach and the analysis of the results.
4. Future directions for our research.

2 Background

In 1965, Nelson coined the word hypertext and defined it as "a body of written or pictorial material interconnected in a complex way that it could not be conveniently represented on a paper. It may contain summaries or maps of its contents and their interrelations; it may contain annotations, additions and footnotes from scholars who have examined it." [Nel65].

A hypertext document, in terms of our project, is made up of nodes and links. Nodes are the actual content of the document and may contain text, figures, tables, audio, video, and other forms of data. Links connect related nodes, where each node is the source or destination of a link. Links can be bi-directional or unidirectional, and links can be typed. Typed links [Tri87] make the relationship between the nodes explicit.

The need for automatic link generation was realized as early as 1945 by Vannevar Bush even before the term hypertext was coined. In his seminal article "As We May Think", Bush describes a device called the Memex that has the capability to connect two related documents, "It affords an immediate step, however, to associative indexing, the basic idea of which is a provision whereby any item may be caused at will to select immediately and automatically another" [Bus45].

Research in automatic hypertext generation uses techniques from pattern matching, information retrieval, natural language processing, and neural networks. Past research deals with the conversion of the marked-up document collections into hypertext. For documents that are not marked-up, pattern matching schemes were used to generate links. Current research deals with the conversion of plain text to hypertext using advanced techniques from information retrieval, natural language processing, and neural networks. A brief overview of the various link generation methodologies are given below.

2.1 Link detection based on pattern matching

In this approach, links are generated using keyword matching. Synonymy and polysemy limit the usefulness of this approach [Tho91]. Synonymy, multiple words with same meaning, is a cause for links being undetected, while polysemy, a single word with multiple meanings, is a cause for the generation of poor links. In addition, keyword-based approaches generate redundant links. For example, consider building links from a document to a dictionary. If a keyword occurs several times within a section, only the first instance of the keyword should be linked to its definition in the dictionary. Otherwise the text will be dominated by the hypertext links and readability of the document will suffer.

2.2 Link detection based on document mark-up

In this approach, the mark-ups in the document are used to generate links [Tho91]. Most links generated by this method are referential in nature. Links can be generated from the references of the object to the actual object. Structural and hierarchical links can also be generated using this approach. For example, a table of contents for a document can be easily generated from the mark-ups of the section and the sub-section headings of the document. A problem with this approach is that not all documents available in digital libraries are marked up.

2.3 Link detection based on information retrieval and visualization

James Allan [All95] describes techniques for link detection and typing using information retrieval (IR) and visualization. Link types are detected by the steps given below.

- Decompose documents into smaller sub-parts – e.g., sections, paragraphs.
- Determine similarity between each sub-part of the first document and each sub-part of the second document. Remember all pairings that have similarity values above a certain threshold.
- Generate links between document pairings that have similarity values above a certain threshold.
- Identify patterns with these links, and use those patterns to describe the type of link.

Though this approach seems promising, it cannot be used interactively due to efficiency considerations [All95], and is therefore not suited for digital libraries. The same is true for other advanced methods such as natural language processing and neural network-based methods. In some cases existing links must be regenerated, requiring the entire link generation process to be repeated, when a document is incrementally added, modified, or deleted. In the next section of this paper we propose a solution for the automation of hypertext generation with the above mentioned problems in mind.

3 Approach

Several problems exist with current hypertext link generation methods: (1) manual construction of hypertext links between related documents in a digital library is expensive [Fox91] [Tho91] [Sal94]; (2) links are often generated by a small number of authors or other users, without knowledge of the breadth of relevant information in the collection; (3) links are generally created once and often precede the addition of new data to the collection; (4) following these links presupposes that what is relevant information for one person is relevant for everyone; (5) the amount of information that will flow into digital libraries is overwhelming to users. Therefore, it is imperative that engines are developed to automatically generate links, based on a system view of the information.

Our system generates hypertext links in an innovative way by using clustering as the basis for link creation. We use the cover-coefficient-based incremental clustering methodology (C^2ICM) [Can93] to generate links between the document (document sub-parts) pairs of each cluster. We use C^2ICM because it can handle large collections [Can95], it is dynamic in nature, and it produces statistically valid clusters compared with those of re-clustering algorithms.

The test collection consists of an ASCII text version of the *ACM Hypertext Compendium* [ACM91] and other heterogeneous documents stored in the PRC *Productivity Edge* document management system, which includes the Virginia Tech research software. Inside a given collection, such as the *ACM Hypertext Compendium*, we assume a structure of three levels: document, paragraph, and sentence. Users of our system can control the indexing to produce vectors at any or all of these levels. We have built our own indexing and subsequent vector processing routines around the Fulcrum *Ful/Text^{TM}* system.

C^2ICM is used to cluster the documents together. The automatic link generation phase is embedded within the clustering phase. Document pairs with a similarity above a given threshold and within each cluster are identified as candidate links. One important attribute of links is the similarity value between the source and destination of the links. Henceforth this attribute will be referred to as link similarity. The *Productivity Edge* database stores the links with other document attributes.

3.1 Incremental clustering and link generation

C^2ICM is a seed-based partitioning type clustering scheme. An advantage of this scheme is that we can predict the number of clusters using the cover-coefficient-based concept. This method of predicting the number of clusters agrees with the hypothesis that the number of clusters within a document collection should be low if the individual documents are dissimilar, and high otherwise. In the case of C^2ICM, the order of document addition does not affect the outcome of the clustering process.

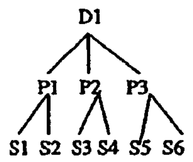
C^2ICM can be broken down into two different phases: the cluster seed selection phase and the cluster construction phase. In the first phase an estimation of the number of clusters and the document seeds is determined using the cover coefficient concept. In the second phase the actual clustering is completed. It is only during the second phase that link generation is completed. The detailed steps in the second phase are explained below.

1. The seed documents from phase 1 are sorted by document number.
2. If this is the first run of the link generation algorithm, then skip to step 7.
3. For each seed document S in the previous clustering structure, if S becomes a non-seed document in this increment, then the cluster containing the document S is falsified.
4. For each new seed S, if S is an old document and S was a non-seed document in the previous cluster structure, then the cluster containing document S is falsified.
5. For each link in the old link set, if the source or the destination of the link is one of the documents (document sub-parts) in the falsified clusters, then the link is deleted.
6. Cluster all documents that belong to the falsified clusters. If document D is added to cluster C, then new links are formed between D and members of the cluster C if they have a similarity value above a certain threshold.
7. Cluster all the new documents with the seed documents. For each document D that is being added to a cluster C, links are formed between D and the members of C if they have similarity value above a certain threshold.

3.2 Illustration of link generation

Figure 1 and Figure 2 illustrate the link generation process. The tree-like structure in the figures has the documents decomposed into paragraphs and sentences. All the documents and document sub-parts are given identifiers. The D's in the figures are document identifiers, the P's are paragraph identifiers, and the S's are sentence identifiers. In this example, both the document (D) and the document sub-parts (P and S) are considered for clustering and link generation. As discussed above, links are generated between pairs in a cluster, with similarity values above a certain threshold.

Document Structure



Cluster and Link Information

Seeds	Non-Seeds
S1	P1, S3, S4, P2
P3	D1, S5, S6
S2	– none –

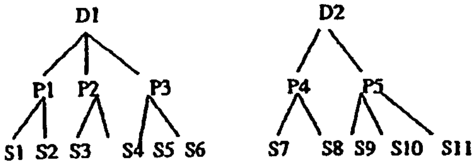
Links:
(S1, S3) (S3, P2) (S5, S6) (P3, S5)

Figure 1: Link Generation: First Increment

Figure 1 shows the first increment of the clustering and link generation process. Document *D1* is added to the collection in this increment. The clustering and the link generation process results in three clusters and four links. The number of links is four because only four document pairs have a similarity value above the specified threshold.

Figure 2 shows the second increment of clustering and link generation. In this increment, document *D2* is added to the collection. During this increment, the non-seed document sub-part *S3*, of the first increment, becomes a new seed and a seed document sub-part *S2*, of the first increment, becomes a non-seed. As a result, the clusters containing both of these document sub-parts are falsified. This results in the deletion of links that are associated with these clusters. The falsified documents (and document sub-parts) and the new documents (and document sub-parts) are then re-clustered. The link generation process results in the dynamic addition of links. Link maintenance and collection browsing are discussed in the next section.

Document Structure



Cluster and Link Information

Seeds	Non-Seeds
S1	P1, P2, S4, S10
S3	S7, S8, P4
P3	D1, S5, S6
S9	S10, S11
P5	D2, S2

Falsified Clusters:
Clusters which has S1, S2 as seeds.

Links deleted:
(S1, S3) (S3, P2)

Added Links:
(S1, S10) (S3, S8) (S9, S11)

Resulting Links:
(S1, S10) (S3, S8) (S5, S6) (S9, S11) (P3, S5)

Figure 2: Link Generation: Second Increment

3.3 Link maintenance

Hypertext links are typically represented in one of three ways: embed all the link information within the document (as in HTML); embed a persistent link id within the document but store the link information externally; or store the link information externally to the document [Dav95]. The last approach has the advantage that the documents need not be modified when links are added or deleted. This approach also supports the navigation of the collection by any browser, since, document formats that are specific to that browser can be built on the fly. Also, this method supports different views of the same collection.

In our system, links are stored externally in a relational database. Our text collection can be navigated using a World-Wide Web browser such as Netscape. Traversal of a link causes the retrieval of a document from *Productivity Edge*, the creation of an HTML document on the fly, and the automatic insertion of links into the document. The next section discusses the usefulness of our link generation approach.

4 Experimental Design and Evaluation

In this section we present two sets of experiments to study the usefulness of the cluster-based link generation process. In the first set of experiments we compare the cluster-based link generation process with an exhaustive method of link generation. By exhaustive, we mean taking a document (document sub-part) in the collection and comparing it to all other documents (document sub-part) in the collection. Links are formed between pairs that have a similarity value above a certain threshold. Links that are detected by the exhaustive approach constitute the complete set. This is because there is no scheme, other than manual, that generates a more complete set of links than the exhaustive method.

In the second set of experiments we compare the cluster-based approach to the manual methods. The manual method of generating links is impractical for large collections. Therefore, this study is performed on a smaller scale than the first set of experiments. This also provides a way to study the quality of links generated by the cluster-based approach.

4.1 Study of link completeness using exhaustive approach

The ratio of the number of links generated by the cluster-based approach to that of the number of links generated by the exhaustive approach can be used as a metric for link completeness. Our hypothesis is that the cluster-based approach should detect most of the links found by the exhaustive approach for higher similarity values between the source and destination of the link. So, we predict that links will be more complete in the higher similarity range than in the lower similarity range. This is because the clustering algorithm groups documents that are more similar to each other in one cluster and our algorithm generates links only within the same cluster.

We ran the experiments on a sample collection of 9 documents from the ASCII version of the "Hypertext Compendium" [ACM91]. The sample documents are listed below:

1. htc1.txt: "A Hypertext Model Supporting Query Mechanisms" by Foto Afrati and Constantinos D. Koutras.
2. htc2.txt: "KMS: A Distributed Hypermedia System for Managing Knowledge in Organizations" by Robert M. Akscyn, Donald L. McCracken and Elise A. Yoder.
3. htc10.txt: "Browsing in Hyperdocuments with the Assistance of a Neural Network" by Frederique Biennier, Michel Guivarch and Jean-Marie Pinon.
4. htc28.txt: "A Retrieval Model for Incorporating Hypertext Links" by W. Bruce Croft and Howard Turtle.
5. htc100.txt: "From Ideas and Arguments to Hyperdocuments: Traveling through Activity Spaces" by Norbert A. Streitz, Jorg Hannemann, and Manfred Thuring.
6. htc102.txt: "A Visual Representation for Knowledge Structures" by Michael Travers.
7. htc108.txt: "Links and structures in hypertext databases for law" by Eve Wilson.
8. htc115.txt: "Hypertext and Information Retrieval: What are the Fundamental Concepts?" by W. Bruce Croft.

9. htcl16.txt: "Hypertext and Higher Education: A Reality Check" by Stephen C. Ehrmann, Steven Erde, Kenneth Morrell, and Ronald F. E. Weissman.

We indexed the collection at three different levels: sentence, paragraph, and document. This resulted in 3633 document and document sub-parts. We then ran the cluster-based scheme and the exhaustive scheme to generate links. The similarity values in our system range from zero to one and the similarity threshold value used for link generation was 0.25. We chose this value because we observed that bad links tend to dominate the link set for lower similarity values.

4.1.1 Experimental design

Let y denote the percentage of links determined by the cluster-based approach when compared to that of the exhaustive link approach. We want to verify that the selection of similarity range has a major impact on the value of y. The other factor that might affect the value of y is the granularity levels of the source and destination nodes of the link. Since we have two factors to study, we have decided to use a 2^2 experiment design [Jai91] to study the effects of the factors on the performance. In this experiment we are interested only in quantifying the relative contribution of the factors to the response variable y.

The nonlinear regression model for this experiment is:

$$y = q_0 + q_A x_A + q_B x_B + q_{AB} x_A x_B \qquad (1)$$

where x's denote the factors, q's denote the effects, y is the response variable, and the subscripts A and B identify the two factors [Jai91].

The experiments will determine the effects of the factors and of their interactions on the response variable. We will also determine the contribution of the factors to the total variation of y, thereby gauging the importance of the factors. The total variation of y, denoted by SST, is related to the squares of the effects by

$$SST = SSA + SSB + SSAB \qquad (2)$$

where

$$SSX = 2^2 q_X{}^2, X \in \{A, B, AB\}. \qquad (3)$$

The fraction of variation explained by a factor X is the ratio SSX/SST.

4.1.2 Factor analysis

The two factors, their corresponding symbols, and their level assignments are shown in Table 1. Since we use a 2^2 factorial design, and the similarity values for the links range from 0.25 to 1.00, we divided this range into two sub-ranges and used them as two different levels for that factor. For the link granularity factor, one choice was to study all the links. For the second level, we chose sentential links. Sentential links are those links where both the source and destination nodes are sentences. In our approach, terms within shorter sentences are generally assigned higher weights, due to the normalized weighting scheme. The effects of this on the response variable can be studied by choosing the second level of the granularity factor to be sentential links.

The sign table and the measured responses are shown in Table 2. In this table E_LINKS denote the number of links that are generated by the exhaustive link generation approach and C_LINKS denote the number of links that are determined by the cluster-based approach. The values of the response variable y are obtained by running the clustering and the exhaustive schemes for four different combinations of the levels of factors A and B.

Table 1: Factors and their levels

Symbol	Factor	Level -1	Level 1
A	Similarity Range	(0.25, 0.6]	(0.6, 1.0)
B	Link Granularity	All links	Sentential links

Table 2: Measured Response

1	A	B	AB	E_LINKS	C_LINKS	y
1	-1	-1	1	10399	1268	12.19
1	1	-1	-1	2132	1223	57.36
1	-1	1	-1	4542	563	12.39
1	1	1	1	1208	565	46.77

The effects, computed using Table 2, are shown in Table 3. Table 3 also contains the percentage of variation explained, which is computed using (2) and (3). The results show that most of the variation of the response is due to the contribution of factor A — the similarity range. This factor contributes 96.6 % to the total variation of the response variable. The granularity level and the interaction of the two factors do not have a comparable effect.

4.1.3 ' Interpretation of the results

The results of our experiments show that the similarity range has a significant impact on the response variable. Also, the result hints that in the higher similarity range the cluster-based algorithm determines a higher percentage of links when compared to the lower similarity range. Table 2 illustrates this fact. Whenever the value of variable A (similarity range) is -1 (lower similarity values), the percentage of links that are determined by the clustering approach is low when compared to the result when the value of A is 1 (higher similarity values). This is true irrespective of the value of B.

To verify the above result, some informal experiments were conducted. We recorded the percentage of links that were determined by the clustering approach for various similarity ranges. Table 4 shows the results of our experiment. This result verifies the outcome of the first experiment. From the table it is clear that as the similarity range increases, the percentage of links detected by the clustering

Table 3: Mean Effects and Allocation of Variation

Effects	Mean Estimate	Variation Explained (%)
q_0	128.72	NA
q_A	79.54	96.61
q_B	10.39	1.6
q_{AB}	10.79	1.74

Table 4: Similarity Ranges vs Link Percentages

Similarity Range	E_LINKS	C_LINKS	Link Percentage
(0.25, 0.3]	4335	403	9.29
(0.3 , 0.4]	4068	434	10.66
(0.4 , 0.5]	1226	222	18.10
(0.5 , 0.6]	770	209	27.14
(0.6 , 0.7]	591	58	9.80
(0.7 , 0.8]	92	35	38.04
(0.8 , 0.9]	47	29	61.70
(0.9 , 1.0]	1492	1101	78.53

Table 5: Link Quality Results

# of manual links	# of cluster links	Link %	Similarity Threshold	# of documents
32	107	34	0.35	15
32	135	43	0.30	15
32	181	43	0.25	15
32	56	41	0.35	3
32	72	41	0.30	3
32	89	43	0.25	3

scheme increases, except for the similarity range (0.6, 0.7]. The higher number of links in the similarity range (0.9, 1.0] can be attributed to the short sentences in the documents. In our study we found that these links are mostly of the referential type. This includes links between the quotes in an article and the actual article, and links between the citations and the reference section of the article. The next section compares the cluster-based link generation method to the manual method of link generation.

4.2 Study of link completeness using manual approach

The purpose of this study is to compare the quality of links generated using our clustering approach with links generated by hand. In the above study of link completeness, our linking method was compared against an exhaustive link generation approach for a relatively small collection. The reasons for selecting an exhaustive approach over a manual approach for a ground truth become quite apparent when a collection consists of hundreds of documents. Since a manual ground truth does not exist for this collection, or any hypertext collection to our knowledge, we chose to manually generate links between only 3 documents for this test. Because of our familiarity with the Hypertext Compendium we knew that the following three documents were similar in content:

1. htc16.txt: "As We May Think" by Vannevar Bush.
2. htc70.txt: "Hypertext: Does It Reduce Cholesterol Too?" by Norm Meyrowitz.
3. htc43.txt: "Information Retrieval From Hypertext: Update.." by Mark E.Frisse et al.

4.2.1 Experiments and observations

The main goal of this test is to compare our approach to the manual approach of link generation and evaluate link quality.

In order to complete this test, we created a full-text collection consisting of 15 documents, including the above documents. We then automatically generated links using the cluster-based approach at a similarity threshold of .35. For the 15 documents, 852 clusters were created and 4,348 links were generated. Our next step was to manually build links from the article "As We May Think" to the other two relevant documents in the collection. We chose Vannevar Bush's article as our base since he is often quoted and cited in the other two documents. The manually generated links from the 3 relevant documents were compared with the links from our clustering approach. Analysis of the results show that 34% of the manually generated links were found by the clustering algorithm. The tests were then run at a similarity level of .30 and .25. In both cases, the number of manual links found improved to 43%.

In the next test we created a collection consisting of only the 3 relevant documents and automatically generated links. This additional test was completed to ensure that variation of the collection size has no impact on the quantity of links generated. For the new collection, 398 clusters were created and 332 links were generated. Comparison of these links with the above method resulted in an improvement to 41% at a similarity level of .35, with a slight decrease in performance to 41% at the .30 level, and performance remaining the same at the .25 level. Table 5 contains the link comparison results for these tests using Vannevar Bush's article as our base (e.g., 32 manual links were created for his article).

4.2.2 Interpretation of results and study of link quality

Careful evaluation of the links missed by our algorithm resulted in the discovery of three common problems: misspellings, document segmentation errors, and misinterpreted punctuation. Differences in the spelling of words such as "Encyclopedia of Britannica" and "Encyclopedia of Brittanica" or "Cabinet Maker" and "Cabinetmaker" between the documents caused several links to be missed. The parser also encountered document segmentation problems with short single lines of text that did not contain proper ending punctuation (i.e., '?' , '!' , or '.'). Often these lines of text were treated as both paragraphs and sentences. The terms in each line generally carry a high weight, causing links to be generated. This occurs most often when section headers are encountered such as "Introduction" or "Section 5". Finally, problems occurred with the breaking up of citations into smaller sentences and difficulties with punctuation embedded in sub-parts such as quotes and ellipses. Where found, misspellings and punctuation errors were corrected in the document, and the tests re-run in order to better evaluate the link generation software – our results reflect these changes. As you will note in Table 5, more links were generated by the automatic method than the manual approach. These additional links were often links to destinations within the same document or links resulting from document segmentation errors. Development of a robust parser will improve the quality of generated links by reducing the number of irrelevant links and by not missing relevant links due to punctuation.

It is our intent to improve the parser and follow up this test with experiments consisting of several users and a larger collection. As we have stated, manual generation of links is very time consuming especially when a document collection is large. The identification of a ground truth collection with relevant links generated between documents at the sentence and paragraph level would be a great asset to our link quality tests.

5 Conclusion and Future Directions

A new method for generating automatic hypertext links was introduced in this paper. This method is both efficient and dynamic to meet the demands of the ever-growing digital library. Document clustering was used as the basis of the automatic link generation process. The link generation process is embedded within the clustering process. The uniqueness is that the clustering technique is applied not only to documents but also to document sub-parts. Links are generated between document pairs that have a similarity value above a certain threshold, and the source and the destination nodes of the links always reside in the same cluster.

Experiments were performed to study link completeness and link quality. For studying link completeness, we compared the cluster-based link generation approach to the exhaustive method of link generation. Results indicate that the cluster-based link generation approach performs better for links that have higher similarity values. In particular, this method works well for referential link types.

In the future, we plan to perform more extensive studies on link quality including comprehensive comparisons of the automatic and the exhaustive link generation approach. Also, we plan to integrate the automatic link generation system with the Envision digital library of Virginia Tech and study how this helps faculty and students who use this library.

References

[ACM91] Association for Computing Machinery, Inc. *The ACM Hypertext Compendium*, ACM Press, 1991.

[All95] Allan, J., *Automatic hypertext construction*, Technical Report TR95-1414, Department of Computer Science, Cornell University, Ithaca, New York, February 1995.

[Bou95] Bournellis, C., *Internet 95*, Internet World, 6(11):47-52, November 1995.

[Bus45] Bush, V., *As we may think*, Atlantic Monthly, 176(1):101-108, 1945.

[Can93] Can, F., *Incremental clustering for dynamic information processing*, ACM Transactions on Information Systems (TOIS), 11(2):143-164, April 1993.

[Can95] Can, F., Fox, E., Snavely, C., & France, R., *Incremental clustering for very large document databases: Initial MARIAN experience*, Information Systems, 84:101-114, 1995.

[Dav95] Davis, H., *To embed or not to embed*. Communications of the ACM, 38(8):108-109, 1995

[Fur89] Furuta, R., Plaisant, C., Shneiderman, B., *Automatically transforming regularly structured text into hypertext*. Electronic Publishing, 2(4):211-229, December 1989.

[Fox91] Fox, E.A., Chen, Q.F., & France, R.K. *Integrating search and retrieval with hypertext*. In Berk, E. & Devlin, J., Eds. Hypertext/Hypermedia Handbook. Pages 329-355, McGraw-Hill Inc., New York, 1991.

[Hea95] Heath, L., Hix, D., Howell, L., Wake, W., Avervoch, G., Labow, E., Guyer , S., Brueni, D., France, R., Dalal, K., Fox, E., *Envision: A user-centered data base of computer science*. Communications of the ACM, 38(4):52-53, 1995.

[Jai91] Jain, R., *The art of computer systems performance analysis*, chapters 12, 17,18, John Wiley and Sons, Inc., New York, 1991.

[Nel65] Nelson, T., *A File structure for the complex, the changing, and the indeterminate*, Proceeding of the ACM 20th National Conference, pages 84-100, 1965.

[Sal94] Salton, G., Allan, J., Buckley, C., Singhal, A. *Automatic analysis, theme generation, and summarization of machine-readable texts*, Science, 264(3):1421-1426, June 1994.

[Tho91] Rearick, T., *Automating the conversion of text into hypertext*. In Berk, E. & Devlin, J., Eds. Hypertext/Hypermedia Handbook. Pages 113-140, McGraw-Hill Inc., New York, 1991.

[Tri87] Trigg, R., Weiser, M., *TEXTNET: A network based approach to text handling*. ACM transactions on office information systems, 4(1):1-23, January 1987.